IN-02
15120
p.31

NASA Technical Memorandum 107644

CFD METHODS DEVELOPMENT CONSIDERATIONS FOR UNSTEADY AERODYNAMIC ANALYSIS

JOHN T. BATINA

JULY 1992

National Aeronautics and Space Administration

Langley Research Center
Hampton, Virginia 23665

CFD METHODS DEVELOPMENT CONSIDERATIONS FOR UNSTEADY AERODYNAMIC ANALYSIS

John T. Batina
NASA Langley Research Center
Hampton, Virginia 23665–5225

Abstract

The development of computational fluid dynamics (CFD) methods for unsteady aerodynamic analysis is described. Special emphasis is placed on considerations that are required for application of the methods to unsteady aerodynamic flow problems. Two broad categories of topics are discussed including grid considerations and algorithm development considerations, and example calculations are presented to illustrate the major points. Although the primary application of these CFD methods is to relatively low-frequency oscillatory phenomena such as flutter, the ideas that are presented may be of value to developers of computational aeroacoustics methods for predicting high–frequency acoustics.

Introduction

Considerable progress in developing computational fluid dynamics (CFD) methods for aerodynamic analysis has been made over the past two decades.[1–3] Although the vast majority of this work has been on the development of methods for steady-state aerodynamic applications, significant progress also has been made in developing CFD methods for unsteady aerodynamic and aeroelastic applications.[2,3] This latter work has been focused primarily on potential flow methods,[4] either at the transonic small-disturbance (TSD) or full-potential equation levels, although research is concentrated currently on developing advanced codes for numerical solution of the Euler or Navier-Stokes equations.[3]

The development of methods for unsteady applications generally has lagged the development of steady methods, primarily because of additional complicating considerations that arise for unsteady applications. Therefore, the purpose of the paper is to describe the development of CFD methods for unsteady aerodynamic analysis with special emphasis on the considerations that are required because of the unsteady application of the methods. These considerations may be divided into two broad categories including grid considerations and algorithm development considerations. In the category of grid considerations, the paper discusses (1) the type of grid, (2) generation details, (3) boundary treatment, and (4) mesh movement. In the category of algorithm development, the paper describes (1) spatial discretizations, (2) temporal discretizations, and (3) adaption techniques. Also, although the primary application of the unsteady aerodynamic methods described herein is to relatively low-frequency oscillatory phenomena such as flutter, the ideas that are presented may be of value to developers of computational aeroacoustics (CAA) methods for predicting high-frequency acoustics.

Grid Considerations

Type of Grid

The first topic in the category of grid considerations is whether a structured or unstructured grid is used.[5] Generally, either type of mesh topology is applicable to steady or unsteady problems, and the use of each has advantages and disadvantages. The majority of work that has been done in CFD over the years has been on developing methods for use on computational grids that have an underlying geometrical structure and thus are referred to as "structured" grids. For example, Fig. 1 shows a structured grid for the NACA 0012 airfoil. The grid is of C-type topology, has 159 points in the wraparound direction, and 49 points in the outward direction. Unsteady applications of methods developed for structured grids generally have been limited to relatively simple geometries such as airfoils, wings, and wing-body configurations.[3] Extensions to more complex configurations often require more sophisticated meshing methodologies such as blocked, patched, chimera, or hybrid type grids. These extensions, in turn, significantly complicate the solution algorithms. Other difficulties arise in moving the grid for unsteady or aeroelastic motion where the grid must conform to the instantaneous shape of the geometry being considered.

An alternative approach is the use of unstructured grids.[6] In two dimensions, these grids are constructed from triangles, and in three dimensions, they consist of an assemblage of tetrahedral cells. The triangles or tetrahedra are oriented in an arbitrary way to conform to the geometry, thus making it possible to treat very complicated shapes. Unsteady aerodynamic and aeroelastic applications of these methods to complete aircraft configurations already have been made.[7] An unstructured grid for the NACA 0012 airfoil is shown in Fig. 2.[8] The total grid has 3300 nodes and 6466 triangles. From the figure it is obvious that the cells are arranged in an arbitrary manner to create a mesh about the airfoil. Figure 3 shows an example of a surface mesh representing the Boeing 747 transport configuration from an unstructured tetrahedral mesh.[9] The total mesh has 101,475 cells for the half-span airplane. There are also over 8000 triangles which lie on the boundaries of the mesh, which include the half-span airplane, the symmetry plane, and the far-field boundaries. The figure demonstrates that additional aircraft components such as pylons and engine nacelles can be modeled easily with an unstructured grid. To further illustrate the point, Fig. 4 shows the details of the grid and steady pressure coefficient contours on the outboard pylon and engine nacelle of the 747 airplane.[9] The pressure coefficients were computed using an Euler code for a freestream Mach number M_∞ of 0.84 and an angle of attack α_0 of 2.73°.

Generation Details

Concerning grid generation details, for steady-state external-flow applications, grids generally are constructed to be fine near the body and coarse near the outer boundaries with some type of stretching in between. This gridding philosophy is reasonably accurate (because of mesh fineness near the body) and efficient (because of few cells in the far field where flow gradients are small) for steady problems but may be inadequate for unsteady problems. For unsteady problems, waves that are propagating through the mesh may be reflected internally if the grid stretches too rapidly.[10–12] To illustrate this problem, Fig. 5 shows an example of internal grid reflections. The calculations were performed for a flat plate airfoil at $M_\infty = 0.85$ using the TSD equation on two Cartesian meshes.[10] The results shown in Fig. 5(a) were obtained using a grid where the points in the vertical direction were determined by an exponential stretching, whereas the results of Fig. 5(b) were obtained using a vertical grid with a more gradual quadratic stretching. As shown in Fig. 5(a), the real and imaginary parts of the unsteady lift-curve slope c_{l_α} as a function of reduced frequency k have oscillations which are due to internal grid reflections. As shown in Fig. 5(b), however, the oscillations have virtually disappeared due to the more gradual stretching

of the vertical grid. From these results, it is evident that careful attention should be paid to the stretching of the grid to alleviate or eliminate internal grid reflections for unsteady aerodynamic applications.

Boundary Treatment

In addition to possible internal grid reflections, waves may also reflect off of the outer boundaries of the mesh, propagate back into the interior of the computational domain, and contaminate the near-field solution.[10] To demonstrate this problem, Fig. 6 shows an example of the effects of boundary reflections. Again the calculations were performed for a flat plate airfoil at $M_\infty = 0.85$ using the TSD equation. As shown in Fig. 6(a), the real and imaginary parts of c_{l_α} have oscillations for low values of k due to boundary reflections. These oscillations, as well as those that occur from internal grid reflections, are of concern because they occur in the range of reduced frequency where the calculations need to be most accurate since this is the frequency range where flutter typically occurs. As shown in Fig. 6(b), however, when so-called nonreflecting far-field boundary conditions[13] are used, the oscillations no longer occur. This is because the nonreflecting conditions tend to absorb waves that are incident on the boundaries. From these results, similar to the treatment of internal grid reflections, it is evident that careful attention also must be given to an accurate treatment of the far-field boundary conditions to alleviate or eliminate boundary reflections.

Mesh Movement

A final grid consideration in the development of CFD methods for unsteady aerodynamic and aeroelastic analysis is how to move or deform the mesh so that it continuously conforms to the instantaneous shape or position of the vehicle. The mesh movement procedure must be general enough to treat realistic aeroelastic motions of complex aircraft configurations. This can be accomplished by modeling the mesh with a network of springs as depicted in Fig. 7.[14] Each edge of each cell is modeled using a spring, where the spring stiffness k_m is inversely proportional to the length of the edge. Points on the outer boundary of the grid are held fixed and the locations of the points on the inner boundary (aircraft) of the grid are specified either through the aeroelastic equations of motion or the prescribed unsteady motion. The displacements of the interior nodes then are determined by solving the static equilibrium equations which result from a summation of forces in each coordinate direction. In practice, these equations are solved using a predictor-corrector procedure. The displacements of the nodes are first predicted using a simple linear extrapolation in time of displacements from previous grids given by[14]

$$\tilde{\delta}_{x_i} = 2\delta^n_{x_i} - \delta^{n-1}_{x_i} \quad \tilde{\delta}_{y_i} = 2\delta^n_{y_i} - \delta^{n-1}_{y_i} \quad \tilde{\delta}_{z_i} = 2\delta^n_{z_i} - \delta^{n-1}_{z_i} \tag{1}$$

The displacements of the nodes are then corrected by solving the static equilibrium equations defined by[14]

$$\delta^{n+1}_{x_i} = \frac{\sum k_m \tilde{\delta}_{x_m}}{\sum k_m} \qquad \delta^{n+1}_{y_i} = \frac{\sum k_m \tilde{\delta}_{y_m}}{\sum k_m} \qquad \delta^{n+1}_{z_i} = \frac{\sum k_m \tilde{\delta}_{z_m}}{\sum k_m} \tag{2}$$

using several Jacobi iterations. To demonstrate mesh movement, instantaneous meshes are presented for a half-span airplane pitched nose up 15° in Fig. 8(a) and pitched nose down 15° in Fig. 8(b). With the spring network the mesh moves smoothly to conform to the instantaneous position of the pitching airplane. The mesh movement procedure using the network of springs is a general method that can treat realistic motions as well as aeroelastic deformations of complex aircraft configurations.

Algorithm Development Considerations

Spatial Discretizations

For either steady or unsteady flow applications, the residual (right-hand-side of the governing fluid flow equations) needs to be discretized in space. Generally speaking, there are two types of spatial discretizations including central differencing and upwind differencing.[9] Either type of differencing has advantages and disadvantages depending upon a number of things such as the problem being solved and the density of the mesh. Specifically, central differencing uses straightforward central differences to approximate all of the spatial derivatives. This type of approach, though, requires explicitly added artificial dissipation terms to include dissipation in the solution. The unsteady Euler equations, for example, are a set of nondissipative hyperbolic partial differential equations which require some form of dissipation to allow convergence to steady state. Furthermore, the explicitly added dissipation terms involve free parameters to control the level of dissipation in the problem. In contrast, upwind differencing accounts for the local wave-propagation characteristics of the flow and thus is naturally dissipative. Consequently, the upwind-approach does not require the adjustment of free parameters to control the dissipation.

Advantages of the central-difference type spatial discretizations are that they are easier to code and take less memory than upwind discretizations.[9] A disadvantage is that they tend to smear shock waves and contact discontinuities and consequently require finer meshes to achieve similar accuracy. Advantages of the upwind-difference type spatial discretizations are that they tend to minimize the artificial dissipation in the problem that is being solved, since they are naturally dissipative, and consequently discontinuities such as shocks and contacts are captured sharply. This attribute of the upwind discretizations in fact may be important for CAA calculations to help ensure that the dissipation is small and thus does not destroy the relatively weak acoustic field. A disadvantage of the upwind methods, however, is that they are generally more difficult to code and require more memory than central-difference methods.

To demonstrate the sharp shock-capturing features of the upwind approach for the spatial discretization of the Euler equations, steady and unsteady results are presented for the NACA 0012 airfoil.[6] Both sets of results were obtained using the flux-difference splitting of Roe. The steady calculation was performed for $M_\infty = 0.8$ and $\alpha_0 = 1.25°$ with the resulting pressure distribution shown in Fig. 9. For this case, there are shock waves on the upper and lower surfaces of the airfoil. The shocks are sharply captured with only one grid point within the shock structure on either surface. Additionally, these sharp shock capturing features of the upwind method carry over to unsteady cases as well. For example, for the NACA 0012 airfoil pitching harmonically at $M_\infty = 0.755$, $\alpha_0 = 0.016°$, with an oscillation amplitude of $\alpha_1 = 2.51°$ at $k = 0.0814$, instantaneous pressure distributions at eight points in time during a cycle of motion are shown in Fig. 10. This is a very interesting case since the shock waves on the upper and lower surfaces of the airfoil periodically appear and disappear during the cycle of motion. It is clear from the results of Fig. 10, that similar to the steady-state example, the calculated shock waves are sharply captured with only one point within the shock structure.

Temporal Discretizations

For unsteady applications, the temporal accuracy and efficiency of the numerical scheme that is used to integrate the governing flow equations are of significant importance. Generally, there are two types of time-integration methods referred to as explicit and implicit.[9] Either type of integration method has advantages and disadvantages depending upon a variety of factors including the type of unsteady problem, the density of the mesh, the characteristic frequency of the problem, etc. Specifically, the most commonly used explicit temporal discretization (for the integration of the Euler or Navier-Stokes equations) is a multi-

stage Runge-Kutta time integration. Codes that are based on a Runge-Kutta integration also typically use local time-stepping, implicit residual smoothing, and multi-grid techniques to accelerate convergence to steady state. Local time-stepping uses the maximum allowable step size at each grid point as determined by a local stability analysis. Implicit residual smoothing permits the use of local time steps that are larger than those required by the CFL (Courant–Friedricks–Lewy) condition. This is accomplished by averaging the residual with values from surrounding grid points. Multi-grid uses corrections that are determined on a sequence of grids of different density to also accelerate convergence to steady state. As for implicit temporal discretizations, factored methods are typically used on structured grids, whereas relaxation procedures, usually of the Gauss-Seidel form, are used on unstructured grids. The implicit discretizations may be time-accurate for unsteady problems and they allow large CFL numbers for rapid convergence to steady state. Generally speaking, codes based on an implicit integration do not require the above-mentioned techniques for convergence to steady-state, although they usually use local time-stepping and sometimes use multi-grid.

A typical four-stage Runge-Kutta time-integration scheme that is used to solve the Euler equations is given by[9]

$$\begin{aligned} Q^{(0)} &= Q^n \\ Q^{(1)} &= Q^{(0)} - \frac{1}{4}\frac{\Delta t}{V} R\left(Q^{(0)}\right) \\ Q^{(2)} &= Q^{(0)} - \frac{1}{3}\frac{\Delta t}{V} R\left(Q^{(1)}\right) \\ Q^{(3)} &= Q^{(0)} - \frac{1}{2}\frac{\Delta t}{V} R\left(Q^{(2)}\right) \\ Q^{(4)} &= Q^{(0)} - \frac{\Delta t}{V} R\left(Q^{(3)}\right) \\ Q^{n+1} &= Q^{(4)} \end{aligned} \tag{3}$$

(A five-stage scheme is typically used to solve the Navier-Stokes equations.) With this type of scheme the residual R at any stage of the scheme is evaluated using the flow variables Q computed in the previous stage. Although the integration constants 1/4, 1/3, 1/2, and 1 are somewhat arbitrarily defined, the last constant must be unity, and the next-to-last constant is 1/2 for second-order temporal accuracy. When time accuracy is not important, such as when marching the equations to steady state, the constants may be defined otherwise, provided that the resulting scheme has good stability and damping properties. Advantages of the explicit temporal discretization such as that represented by Eq. (3) are that it is simple to code and takes less memory than an implicit time integration. A disadvantage, however, is that the explicit method generally is inefficient for unsteady problems because of the very small time steps that are required for numerical stability.

A typical implicit temporal discretization for structured grids is a three-factor, spatially-split method given by

$$\left[I + \frac{\Delta t}{\mathrm{vol}}\delta_x A\right]\left[I + \frac{\Delta t}{\mathrm{vol}}\delta_y B\right]\left[I + \frac{\Delta t}{\mathrm{vol}}\delta_z C\right]\Delta Q = \frac{-\Delta t}{\mathrm{vol}} R \tag{4}$$

Equation (4) is written in the so-called delta form where ΔQ represents the change in flow variables from one time step to the next. The equation as written is first-order-accurate in time (a second-order-accurate version simply adds a time derivative term to the right-hand side) and is efficient because each of the implicit factors on the left-hand side of the equation involves a spatial derivative in only one coordinate direction. The equation is solved by performing three sweeps through the mesh, each of which uses one of the implicit factors.

A typical implicit temporal discretization for unstructured grids is a Gauss-Seidel relaxation procedure given by[9]

$$\left[\frac{\text{vol}}{\Delta t}I + \sum_{m=1}^{4} A^{+}(Q_j)\Delta S\right]\Delta Q_j + \sum_{m=1}^{4} A^{-}(Q_m)\Delta S\Delta Q_m = -R \tag{5}$$

which is also first-order accurate in time. The relaxation procedure is implemented by first ordering the cells that make up the unstructured mesh, usually from upstream to downstream. Equation (5) is then solved by performing two sweeps through the mesh, one in the forward direction and one in the backward direction. As the sweeps are performed, the equation is solved directly for ΔQ_j while the values for ΔQ_m are updated in a Gauss-Seidel fashion.

Either implicit method (Eq. (4) or Eq. (5)) requires the calculation of the flux jacobian matrices represented by A, B, and C in Eq. (4) and by A^+ and A^- in Eq. (5). These matrices are defined as the derivative of the respective flux that appears in the residual by the conserved variables Q. Advantages of the implicit temporal discretizations are that they are numerically stable for large CFL numbers and consequently enable rapid convergence to steady state.[9] Furthermore for unsteady applications, they allow the selection of the step size based on the physical problem that is being solved rather than on numerical stability considerations. A disadvantage though, is that they require more memory than an explicit method, primarily due to having to store the flux jacobians. Also, the linearization and either factorization (Eq. (4)) or relaxation (Eq. (5)) errors associated with the implicit methods may be too large for a given step size and thus contaminate the solution. To illustrate this problem, Fig. 11 shows the effects of step size on instantaneous pressure distributions using the Gauss-Seidel relaxation procedure of Eq. (5).[8] The calculations are for the same pitching NACA 0012 airfoil case presented in Fig. 10. Here though, three sets of results were obtained corresponding to using 250, 1000, and 2500 steps per cycle of motion. The figure shows the instantaneous pressure distribution at one instant in time (instantaneous angle of attack $\alpha(\tau)$ equal to 2.34° which is 69° ($k\tau$) into a cycle of motion). These results indicate that large errors in the strengths and locations of the shock waves on the upper and lower surfaces of the airfoil can occur when too large of a step size is used (corresponding to 250 steps per cycle). However, when an appropriately small step size is used (corresponding to 2500 steps per cycle), the correct solution is obtained with a shock of moderate strength on the upper surface and subcritical flow (no shock wave) about the lower surface.

Adaption Techniques

Other considerations in the category of algorithm development involve spatial[15] and temporal[16] adaption techniques. These techniques are usually implemented on unstructured grids to produce solutions of high spatial and temporal accuracy at minimal computational cost. The procedures are applicable to either steady or unsteady aerodynamic problems, although for the unsteady case, special attention is required to ensure the time-accuracy of the solution. Also, although temporal adaption is normally applied only to unsteady cases, that need not necessarily be the case.

As for spatial adaption,[15] the technique involves an enrichment procedure to add points in high gradient regions of the flow. The technique also involves a coarsening procedure to remove points where they are not needed. This is accomplished by "marking" cells for either enrichment or coarsening using some type of error indicator to judge the local accuracy of the solution. The objective is to produce solutions of high spatial accuracy at minimal computational cost by simply minimizing the total number of cells in the grid. Figures 12(a) and 12(b) show the various combinations of cells that are possible for an unstructured grid of triangles when using enrichment and coarsening procedures, respectively. It is noted

that in three dimensions, when using unstructured grids of tetrahedral cells, many more combinations of enrichment and coarsening are possible, all of which must be accounted for in the coding of the spatial adaption techniques.

As an example of mesh enrichment for a steady flow,[5] Fig. 13 shows conical vortex-dominated (Euler) flow solutions for a 75° swept flat plate delta wing at a supersonic freestream Mach number of 1.4. The wing is at 20° angle of attack and has 10° of yaw. The solution was obtained by adapting the original coarse mesh of Fig. 13(a) three times to the instantaneous flow. The final result shown in Fig. 13(b) is a highly accurate solution of the conical Euler equations, produced by using an order of magnitude fewer grid points than if a globally fine mesh were used. To demonstrate the spatial adaption procedures for an unsteady case, results were again obtained for the pitching NACA 0012 airfoil case presented previously. Figure 14(a) shows the instantaneous adapted meshes obtained using three levels of enrichment on a coarse background mesh and Fig. 14(b) shows the corresponding instantaneous density contour lines ($\Delta\rho = 0.02$). The instantaneous meshes and density contours are plotted at the same eight points in time as before. The meshes (Fig. 14(a)) clearly indicate the enrichment in regions near the shock waves and near the stagnation points. They also show coarsened regions where previously enriched regions have relatively small flow gradients. The density contours during the cycle (Fig. 14(b)) demonstrate the ability of the spatial adaption procedures to produce sharp transient shock waves. The results were obtained with a computational savings of a factor of seventeen in comparison to using a globally fine mesh for the same case.

Analogous to spatial adaption, temporal adaption[16] may be used to improve the computational efficiency of explicit time-integration methods for unsteady aerodynamic applications. Temporal adaption can be thought of as time-accurate local time-stepping where each grid cell is integrated according to the local flow physics and numerical stability. The efficiency of the method comes from using small time steps where they are needed and large time steps where they are not. The "trick" is to accomplish this in a time-accurate manner. Simply stated, the method integrates small cells with small time steps and large cells with large time steps, as depicted in Fig. 15. Time accuracy is maintained by bringing all of the cells to the same time level as dictated by the step size of the largest cell.

To demonstrate the temporal adaption procedure, results were once more obtained for the same pitching NACA 0012 airfoil problem as before.[16] Figure 16 shows calculated results obtained using temporal adaption and global time-stepping as well as comparisons with the experimental pressure data of Ref. 17. The two sets of calculated pressures agree very well with each other. This excellent agreement verifies the time accuracy of the solution computed using temporal adaption, which was obtained at one-fourth of the CPU time that the global time-stepping solution required. Also, both sets of calculated results agree reasonably well with the experimental data.

Concluding Remarks

The development of CFD methods for unsteady aerodynamic analysis was described. Special emphasis was placed on considerations that are required for application of the methods to unsteady aerodynamic flow problems. Two broad categories of topics were discussed including grid considerations and algorithm development considerations, and example calculations were presented to illustrate the major points. Although the primary application of these methods is to relatively low-frequency oscillatory phenomena such as flutter, the ideas that were presented may be of value to developers of CAA methods for predicting high-frequency acoustics.

This page left intentionally blank.

References

1. Jameson, A.: "Successes and Challenges in Computational Aerodynamics," AIAA Paper No. 87–1184, January 1987.
2. Edwards, J. W.; and Thomas, J. L.: "Computational Methods for Unsteady Transonic Flows," AIAA Paper No. 87–0107, January 1987.
3. Edwards, J. W.; and Malone, J. B.: "Current Status of Computational Methods for Transonic Unsteady Aerodynamic and Aeroelastic Applications," Paper No. 1, presented at the AGARD Structures and Materials Panel Specialist's Meeting on Transonic Unsteady Aerodynamics and Aeroelasticity, San Diego, California, October 9–11, 1991.
4. Edwards, J. W.: "Applications of Potential Theory Computations to Transonic Aeroelasticity," ICAS-86–2.9.1, September 1986.
5. Batina, J. T.: "Unsteady Aerodynamics Methods for Transonic Aeroelastic Analysis," Paper No. 89–002, presented at the European Forum on Aeroelasticity and Structural Dynamics 1989, Aachen, Germany, April 17–19, 1989.
6. Batina, J. T.; Lee, E. M.; Kleb, W. L.; and Rausch, R. D.: "Unstructured-Grid Methods Development for Unsteady Aerodynamic and Aeroelastic Analyses," Paper No. 2, presented at the AGARD Structures and Materials Panel Specialist's Meeting on Transonic Unsteady Aerodynamics and Aeroelasticity, San Diego, California, October 9–11, 1991.
7. Rausch, R. D.; Batina, J. T.; and Yang, H. T. Y.: "Three-Dimensional Time-Marching Aeroelastic Analyses Using an Unstructured-Grid Euler Method," AIAA Paper No. 92–2506, April 1992.
8. Batina, J. T.: "Implicit Flux-Split Euler Schemes for Unsteady Aerodynamic Analysis Involving Unstructured Dynamic Meshes," AIAA Paper No. 90–0936, April 1990.
9. Batina, J. T.: "Unstructured-Grid Methods Development-Lessons Learned," presented at the 4th International Symposium on Computational Fluid Dynamics, Davis, California, September 9–12, 1991.
10. Seidel, D. A.; Bennett, R. M.; and Whitlow, W., Jr.: "An Exploratory Study of Finite Difference Grids for Transonic Unsteady Aerodynamics," NASA TM 84583, December 1982.
11. Bland, S. R.: "Personal Computer Study of Finite-Difference Methods for the Transonic Small Disturbance Equations," NASA TM 102582, December 1989.
12. Bland, S. R.: "Suggestions for CAP-TSD Mesh and Time Step Input Parameters," NASA TM 104083, June 1991.
13. Whitlow, W., Jr.: "Characteristic Boundary Conditions for Three-Dimensional Transonic Unsteady Aerodynamics," NASA TM 86292, October 1984.
14. Batina, J. T.: "Unsteady Euler Algorithm With Unstructured Dynamic Mesh for Complex-Aircraft Aeroelastic Analysis," AIAA Paper No. 89–1189, April 1989.
15. Rausch, R. D.; Batina, J. T.; and Yang, H. T. Y.: "Spatial Adaption Procedures on Unstructured Meshes for Accurate Unsteady Aerodynamic Flow Computation," AIAA Paper No. 91–1106, April 1991.
16. Kleb, W. L.; Batina, J. T.; and Williams, M. H.: "Temporal Adaptive Euler/Navier-Stokes Algorithm for Unsteady Aerodynamic Analysis of Airfoils Using Unstructured Dynamic Meshes," AIAA Paper No. 90–1650, June 1990.
17. Landon, R. H.: "NACA 0012. Oscillating and Transient Pitching," Data Set 3 in AGARD-R-702, Compendium of Unsteady Aerodynamic Measurements, August 1982.

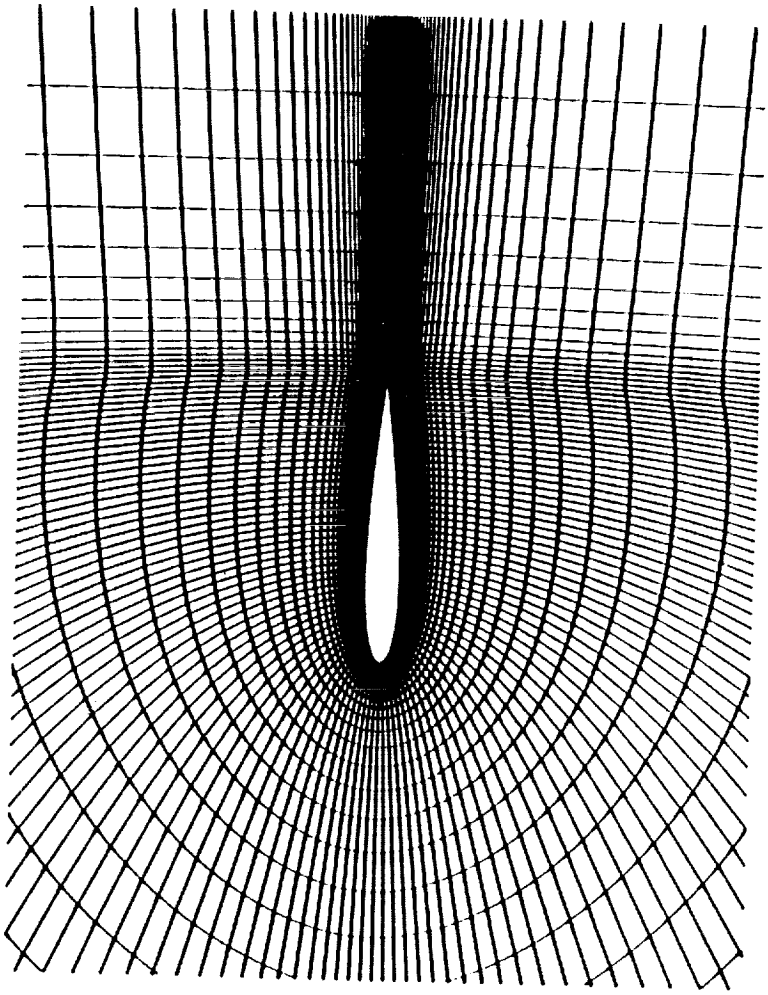


Fig. 1 Structured grid about the NACA 0012 airfoil.

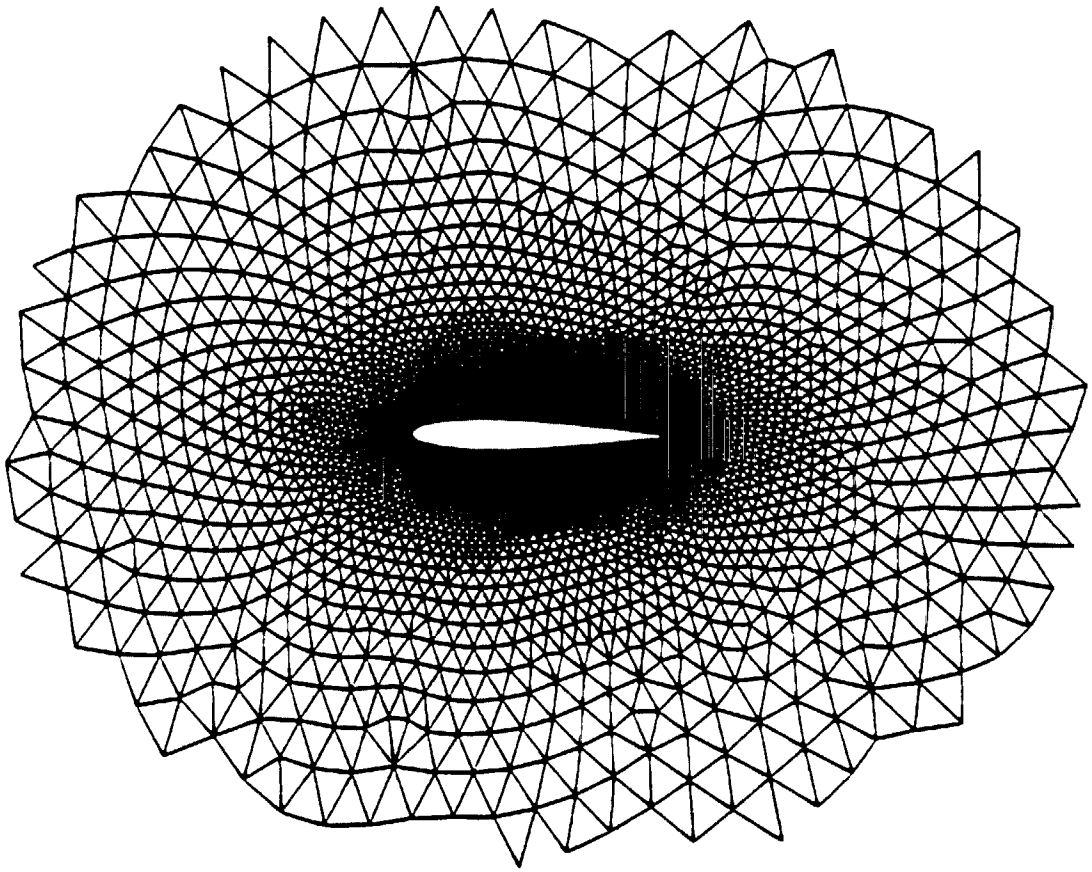


Fig. 2 Unstructured grid about the NACA 0012 airfoil.

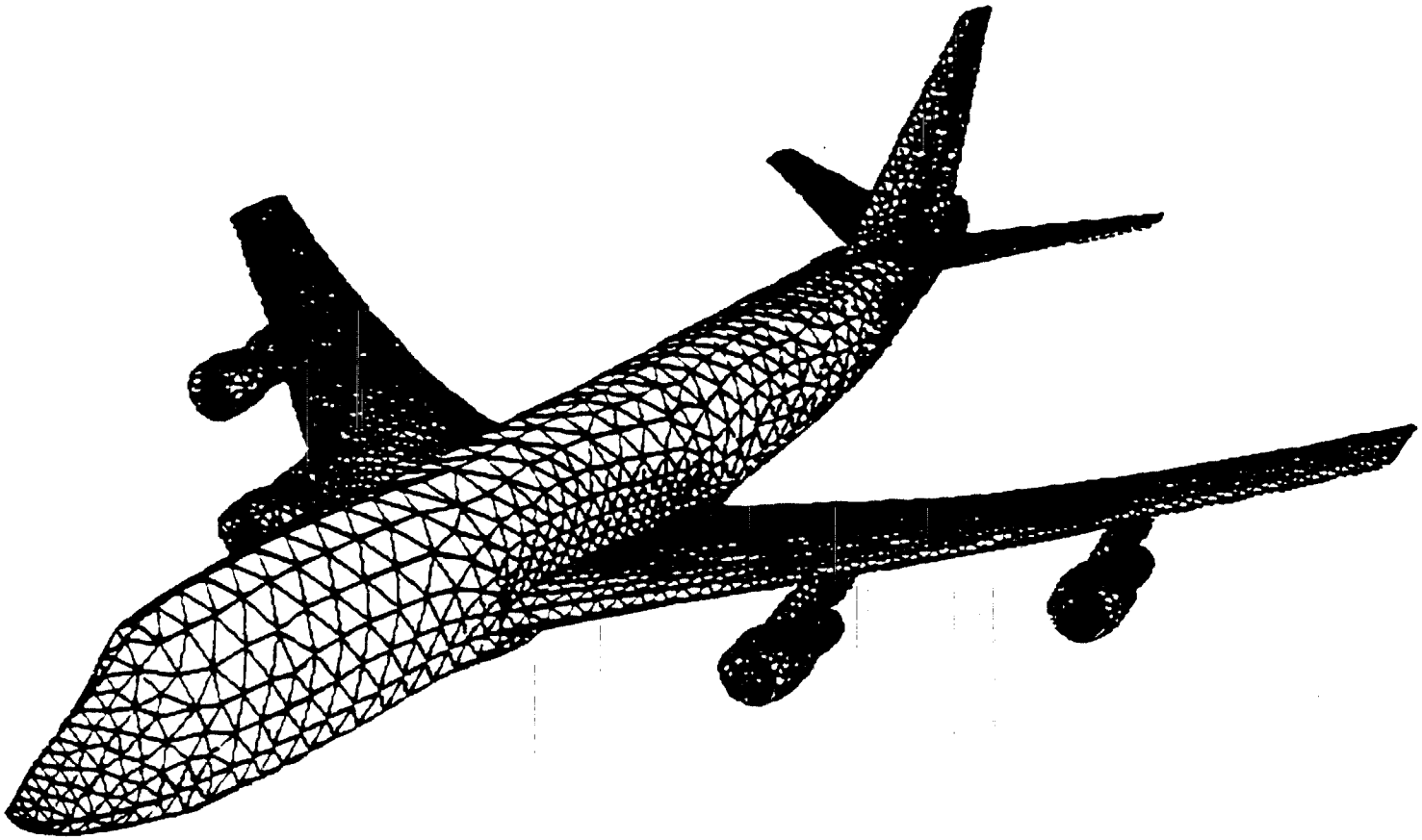


Fig. 3 Surface mesh from an unstructured tetrahedral mesh about the Boeing 747 transport configuration.

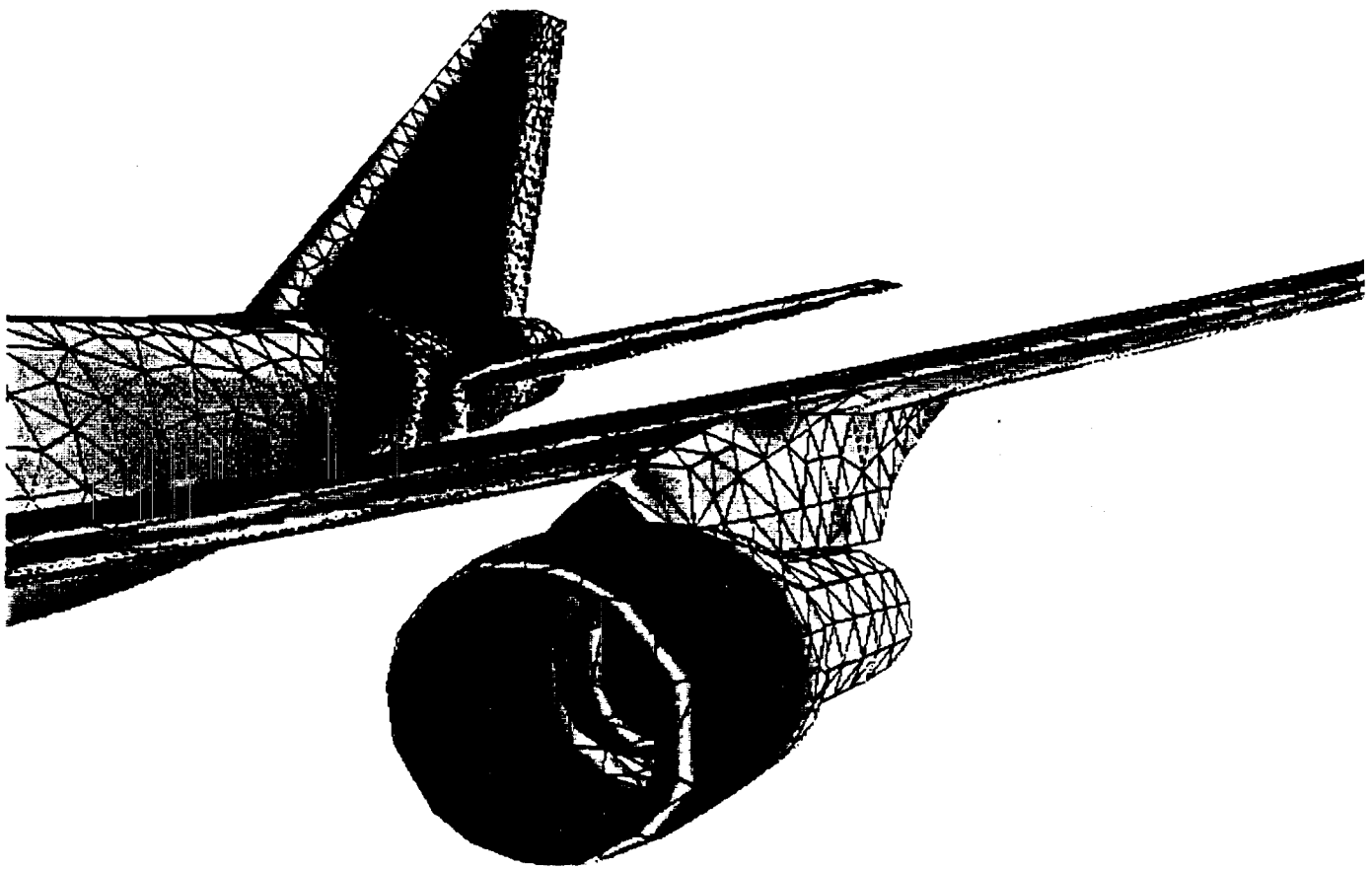
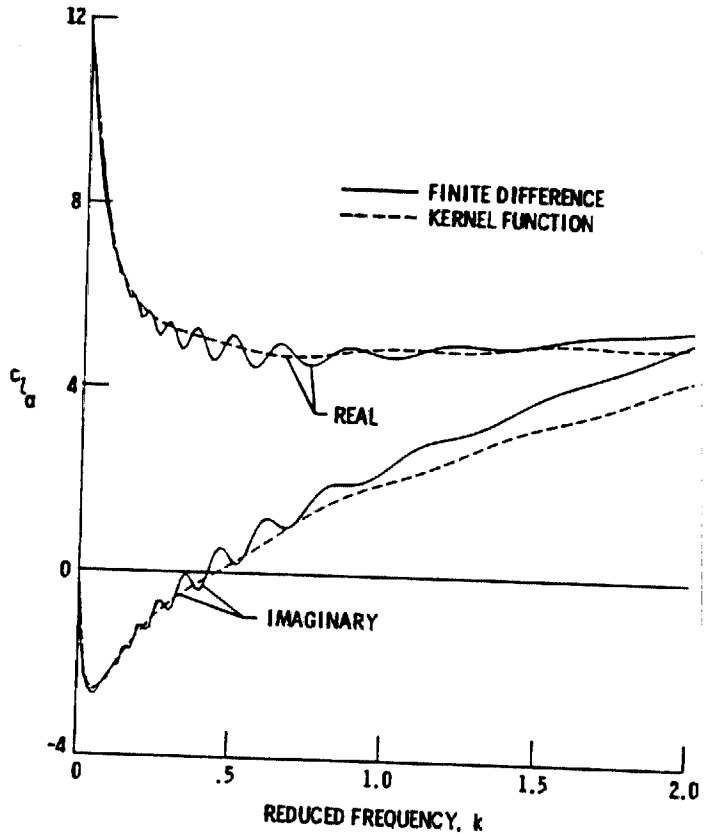
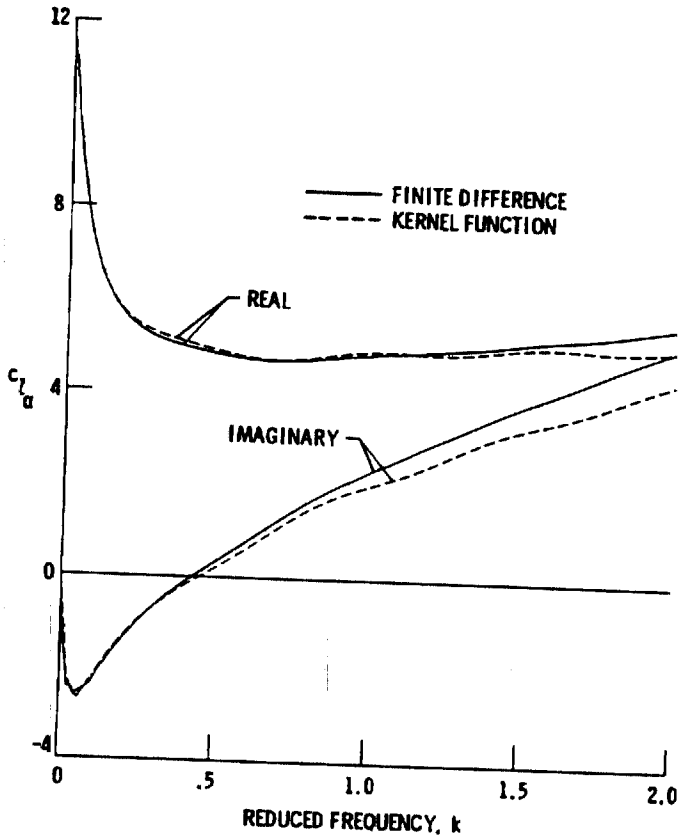


Fig. 4 Details of surface grid and steady pressure contours on the outboard pylon and engine nacelle of the Boeing 747 transport configuration computed using the Euler equations at $M_\infty = 0.84$ and $\alpha_0 = 2.73°$.

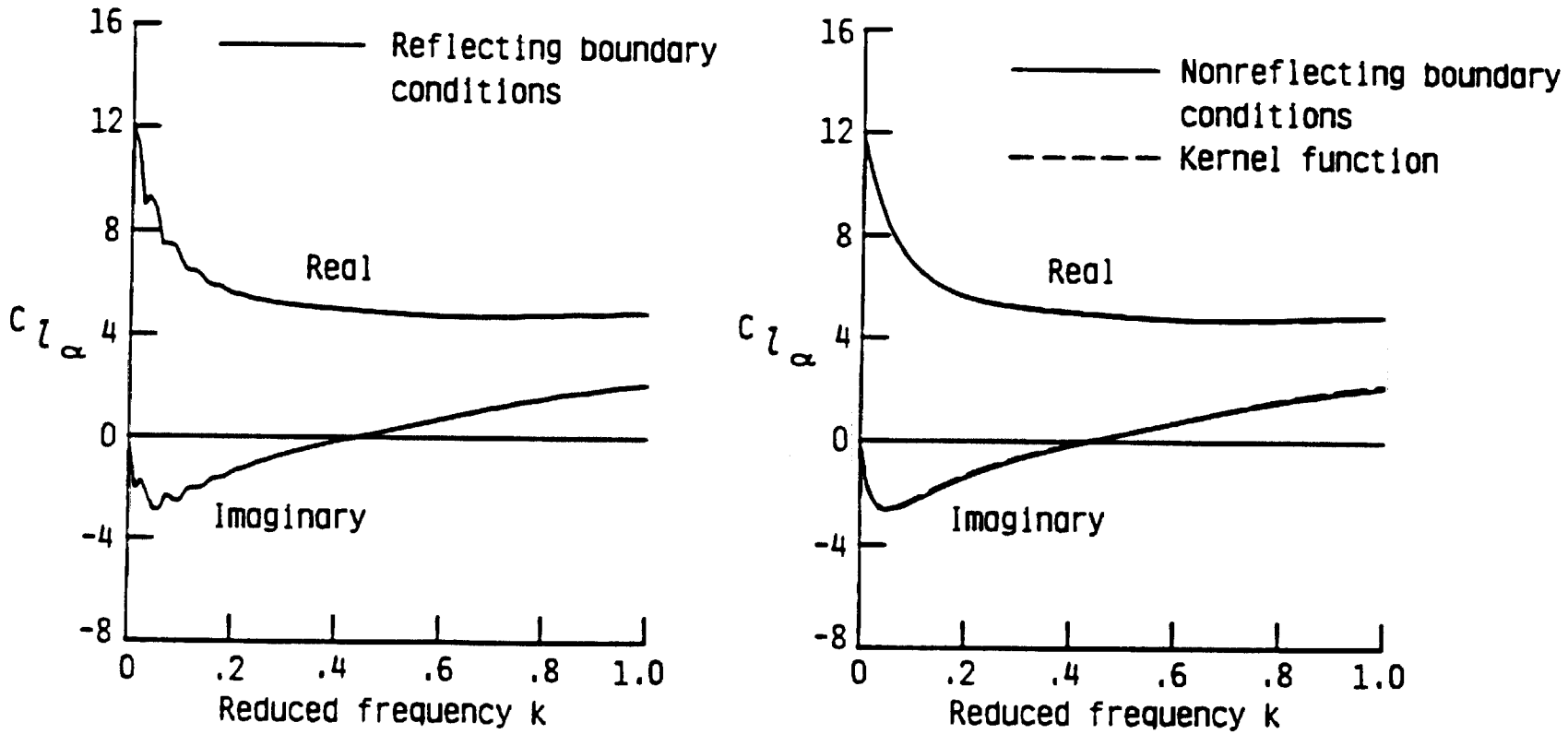


(a) exponential stretching.



(b) quadratic stretching.

Fig. 5 Effects of grid stretching on the unsteady lift-curve slope of a flat plate airfoil at $M_\infty = 0.85$ and $\alpha_0 = 0°$ computed using the TSD equation on Cartesian meshes.



(a) reflecting boundary conditions.

(b) nonreflecting boundary conditions.

Fig. 6 Effects of far-field boundary conditions on the unsteady lift-curve slope of a flat plate airfoil at $M_\infty = 0.85$ and $\alpha_0 = 0°$ computed using the TSD equation.

This page left intentionally blank.

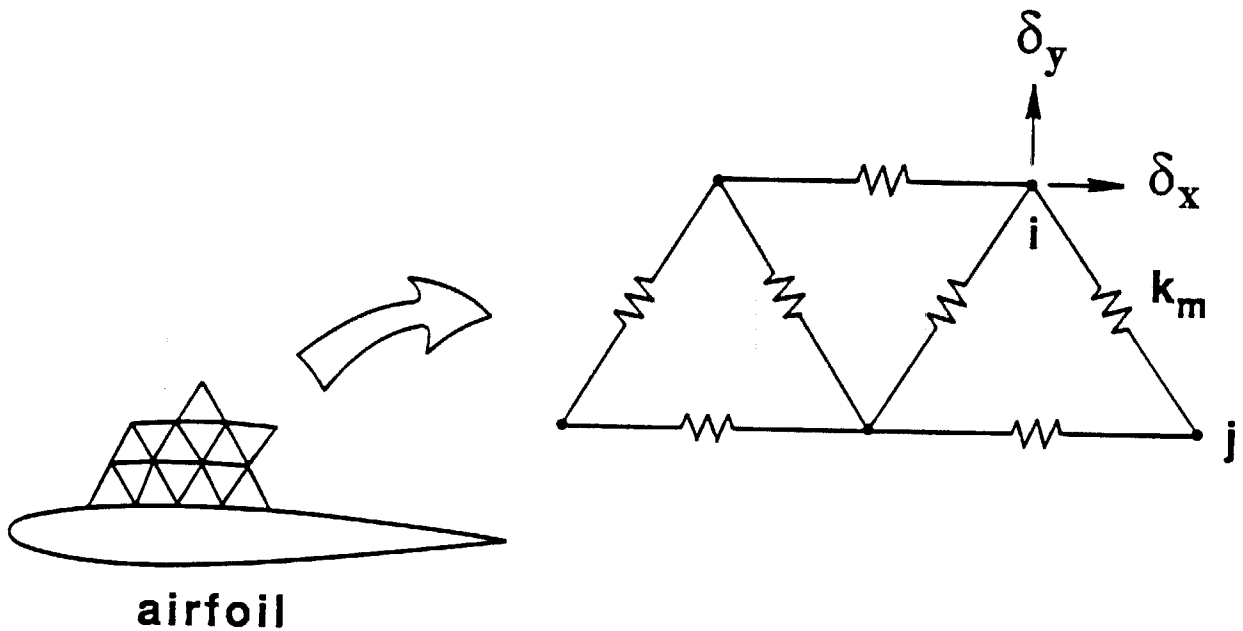
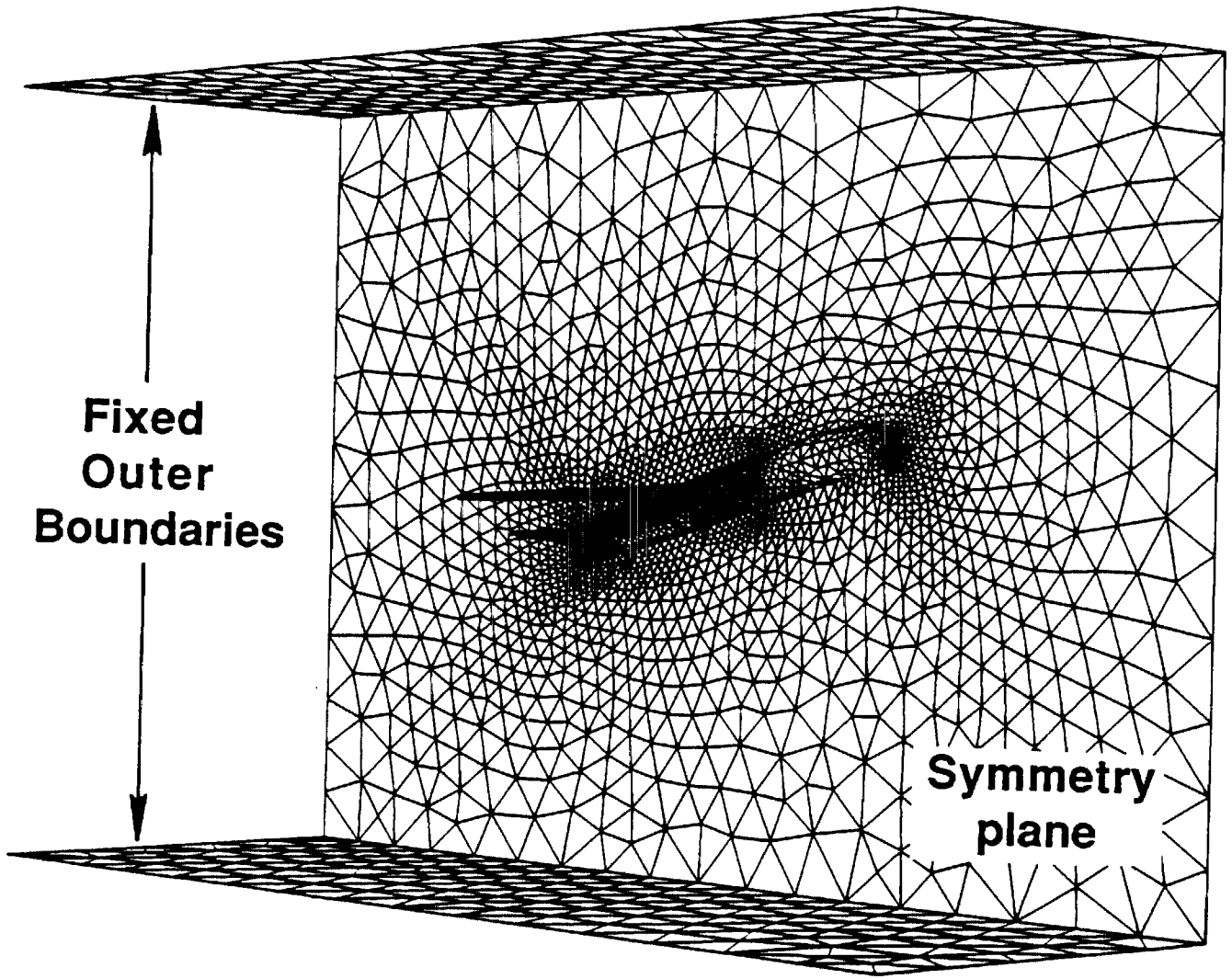
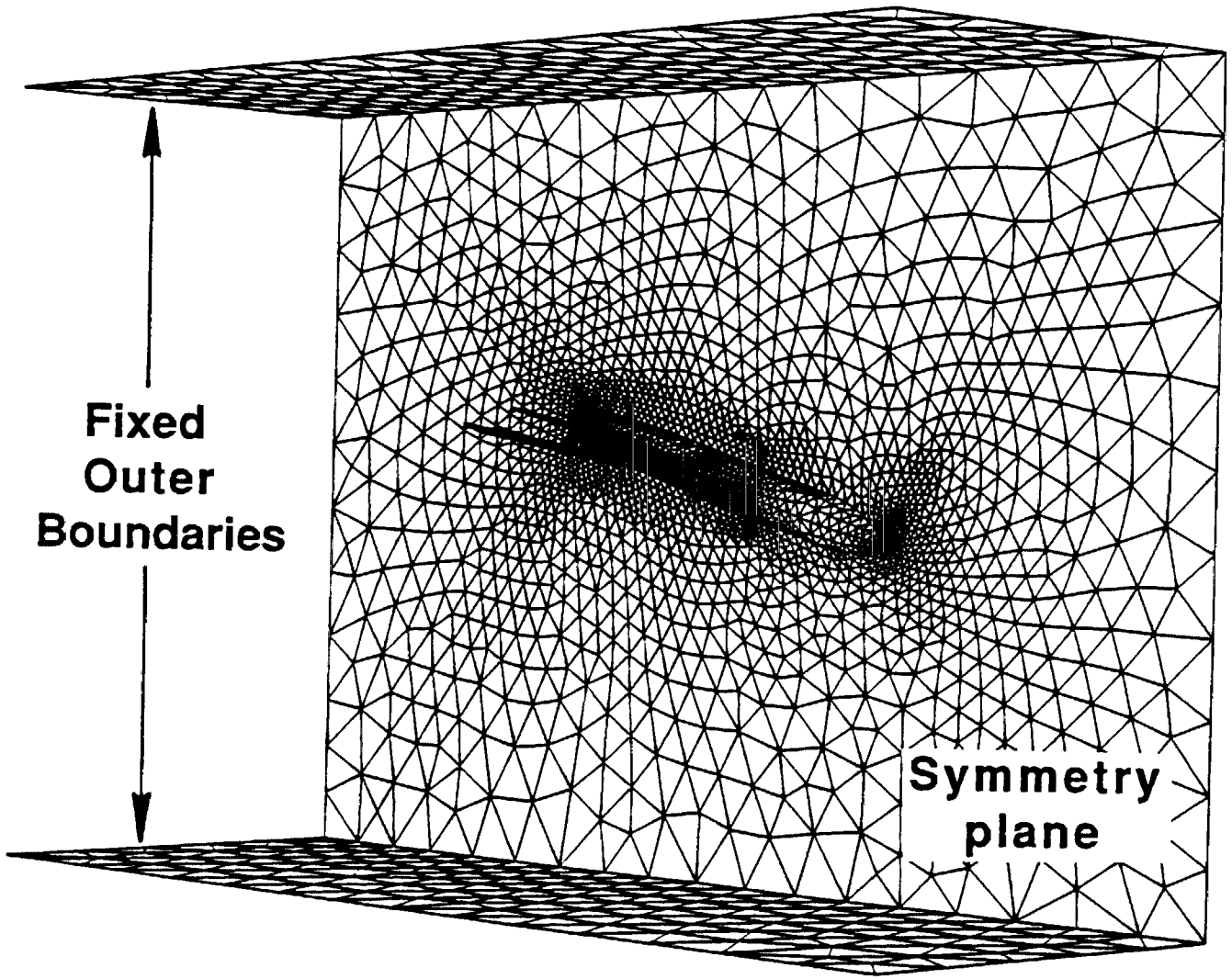


Fig. 7 Mesh modeling by a network of springs.



(a) nose up 15°.

Fig. 8 Instantaneous meshes for a pitching airplane.



(b) nose down 15°.

Fig. 8 Concluded.

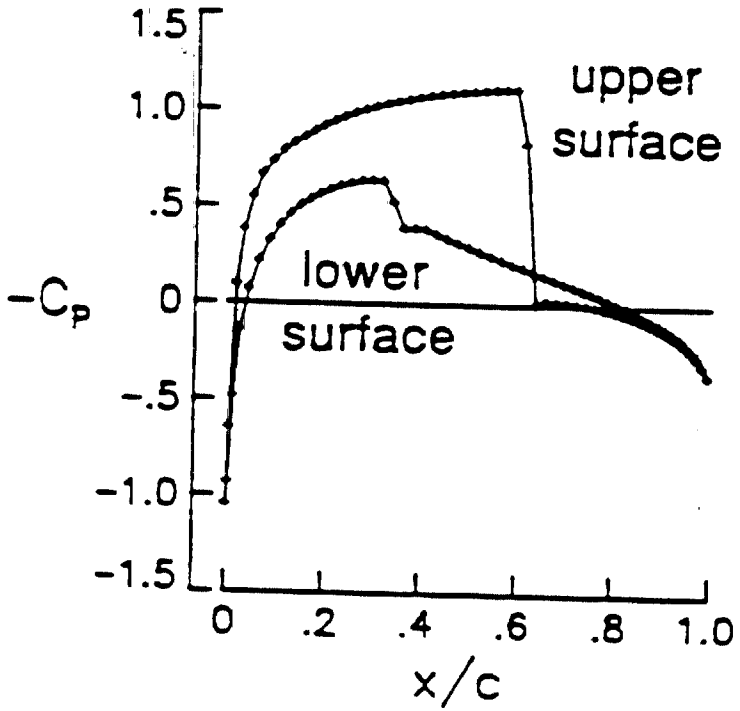


Fig. 9 Steady pressure distribution for the NACA 0012 airfoil at $M_\infty = 0.8$ and $\alpha_0 = 1.25°$ computed using an upwind-type Euler solution algorithm with flux-difference splitting.

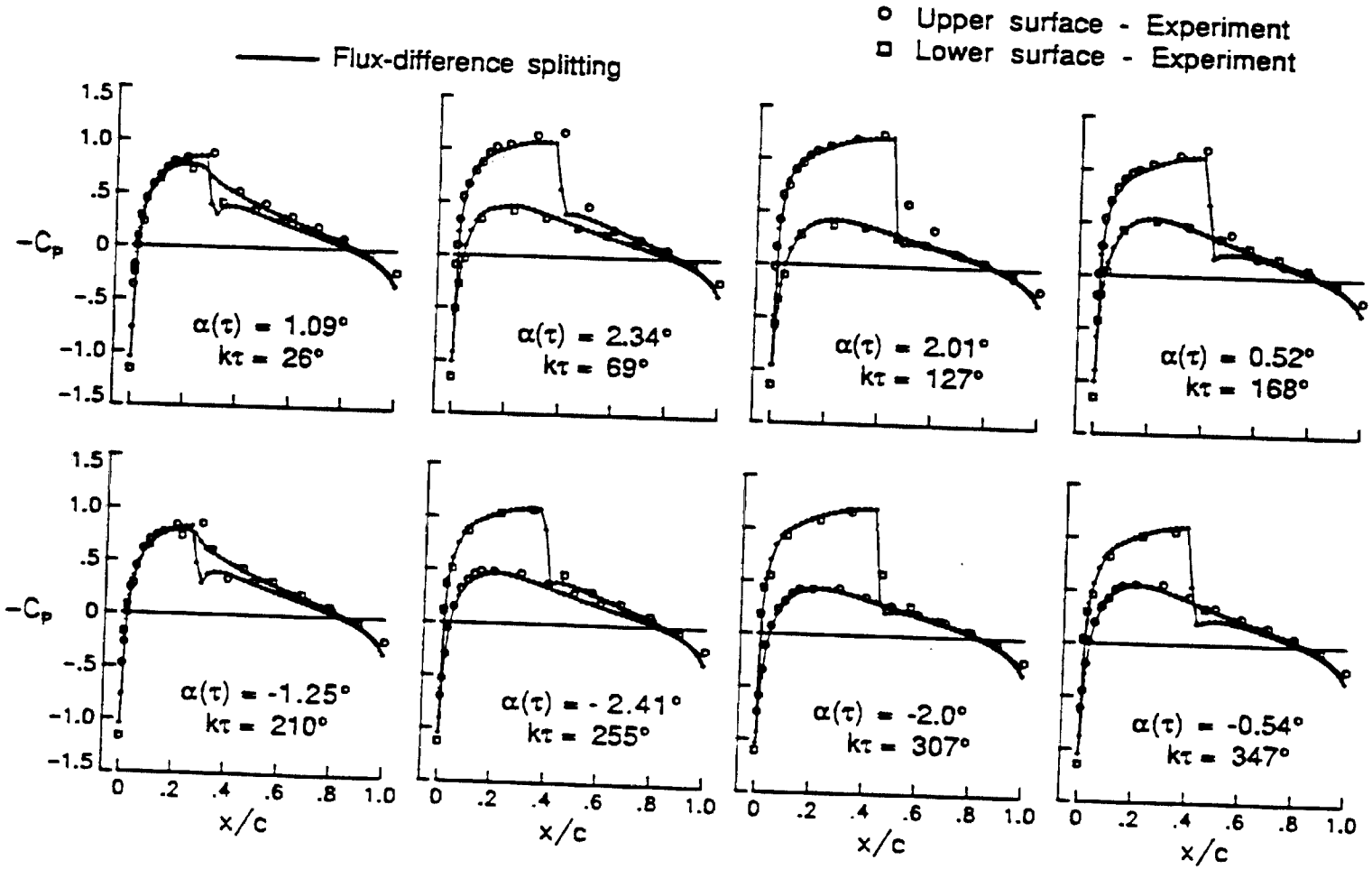


Fig. 10 Instantaneous pressure distributions for the NACA 0012 airfoil pitching harmonically at $M_\infty = 0.755$, $\alpha_0 = 0.016°$, $\alpha_1 = 2.51°$, and $k = 0.0814$ computed using an upwind-type Euler solution algorithm with flux-difference splitting.

This page left intentionally blank.

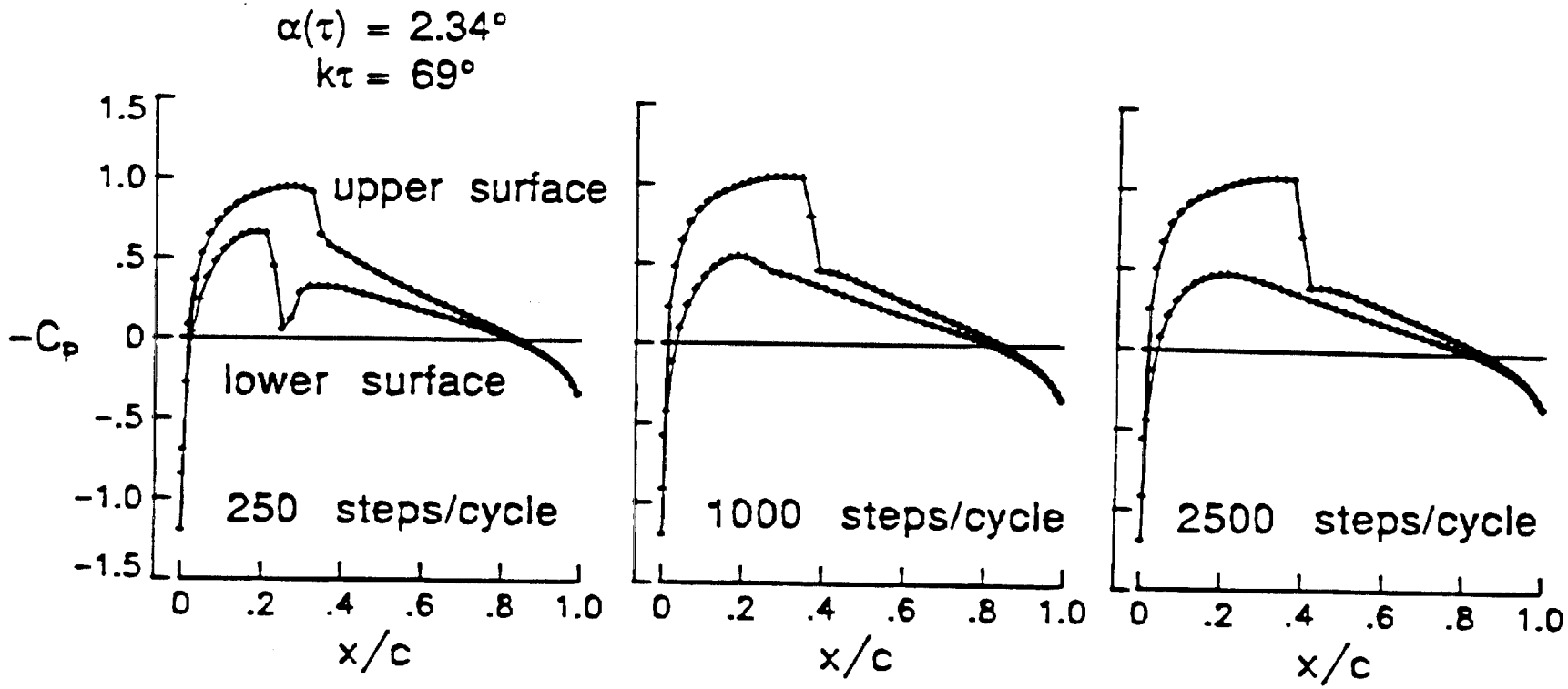
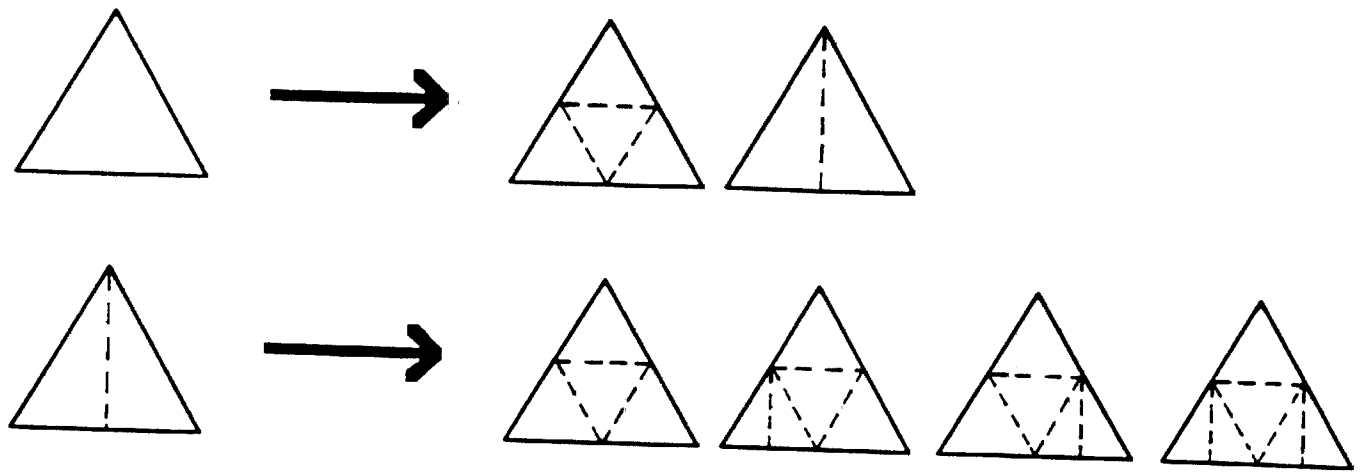
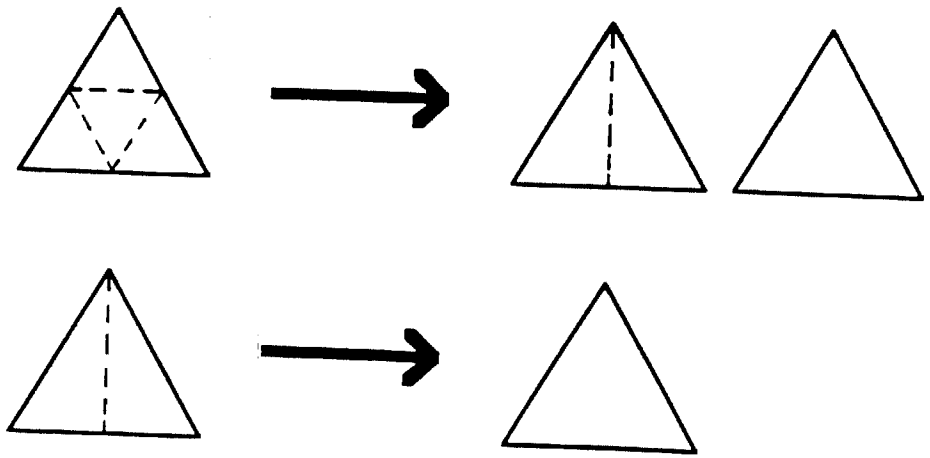


Fig. 11 Effects of step size on the instantaneous pressure distribution at $k\tau = 69°$ (corresponding to $\alpha(\tau) = 2.34°$) for the NACA 0012 airfoil pitching harmonically at $M_\infty = 0.755$, $\alpha_0 = 0.016°$, $\alpha_1 = 2.51°$, and $k = 0.0814$ computed using an implicit Euler solution algorithm.

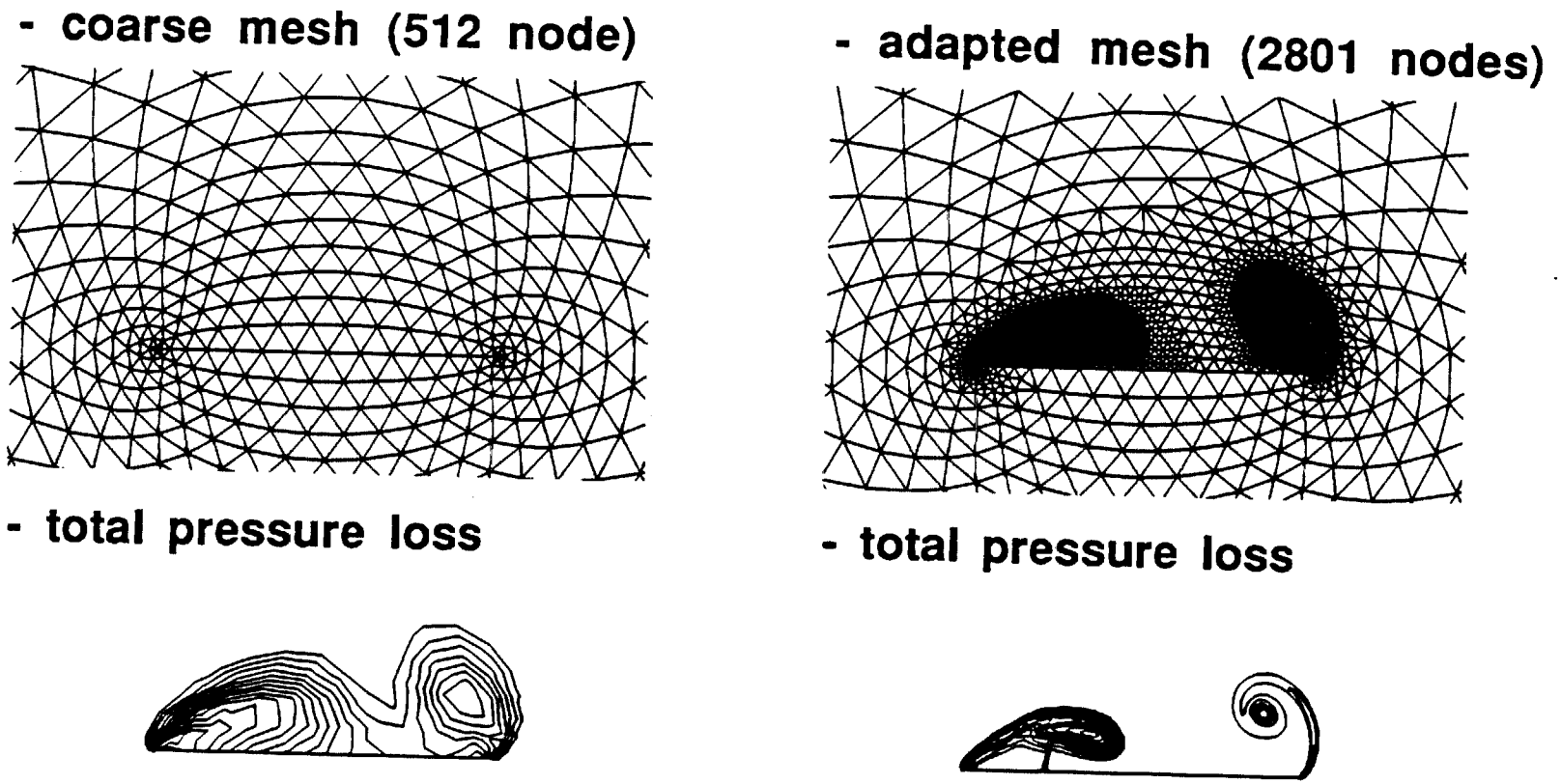


(a) enrichment possibilities.



(b) coarsening possibilities.

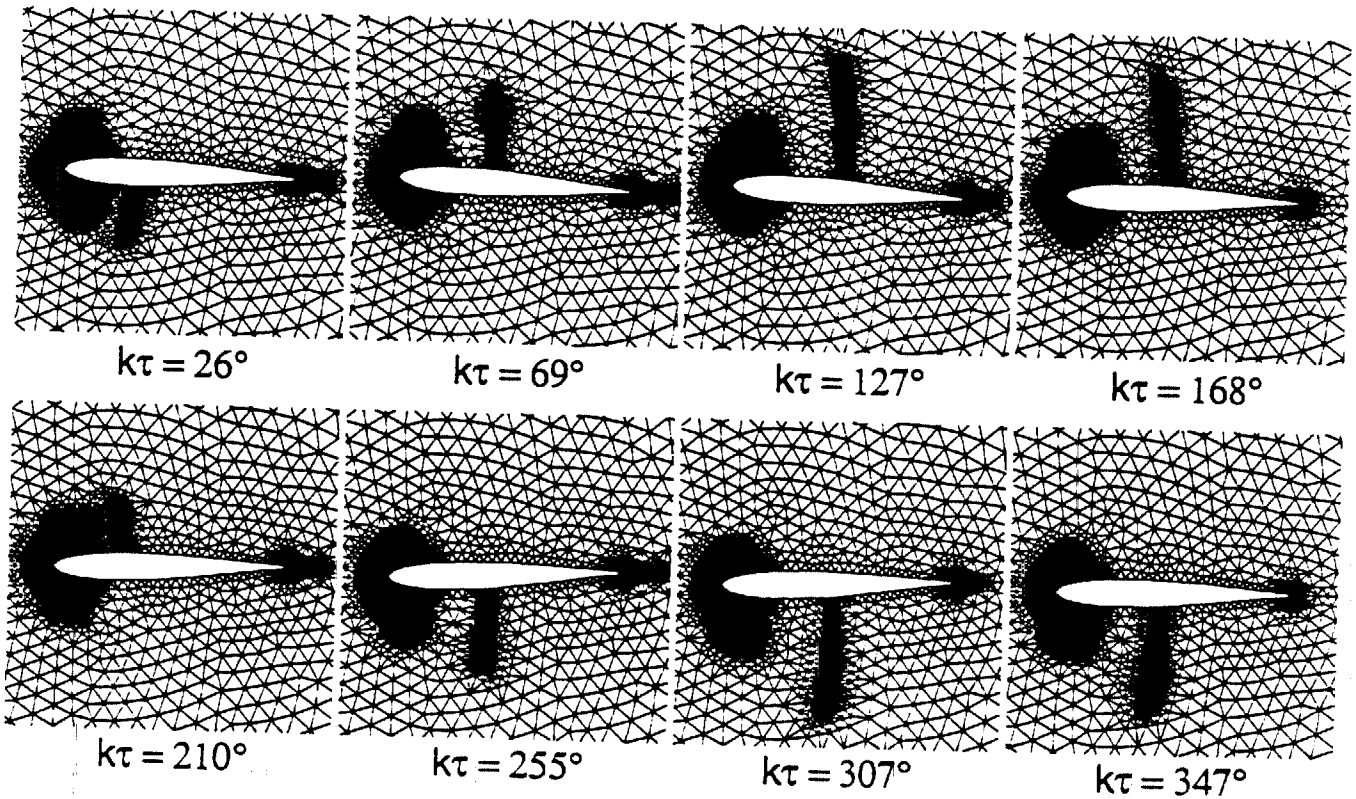
Fig. 12 Details of the spatial adaption procedures.



(a) starting solution.

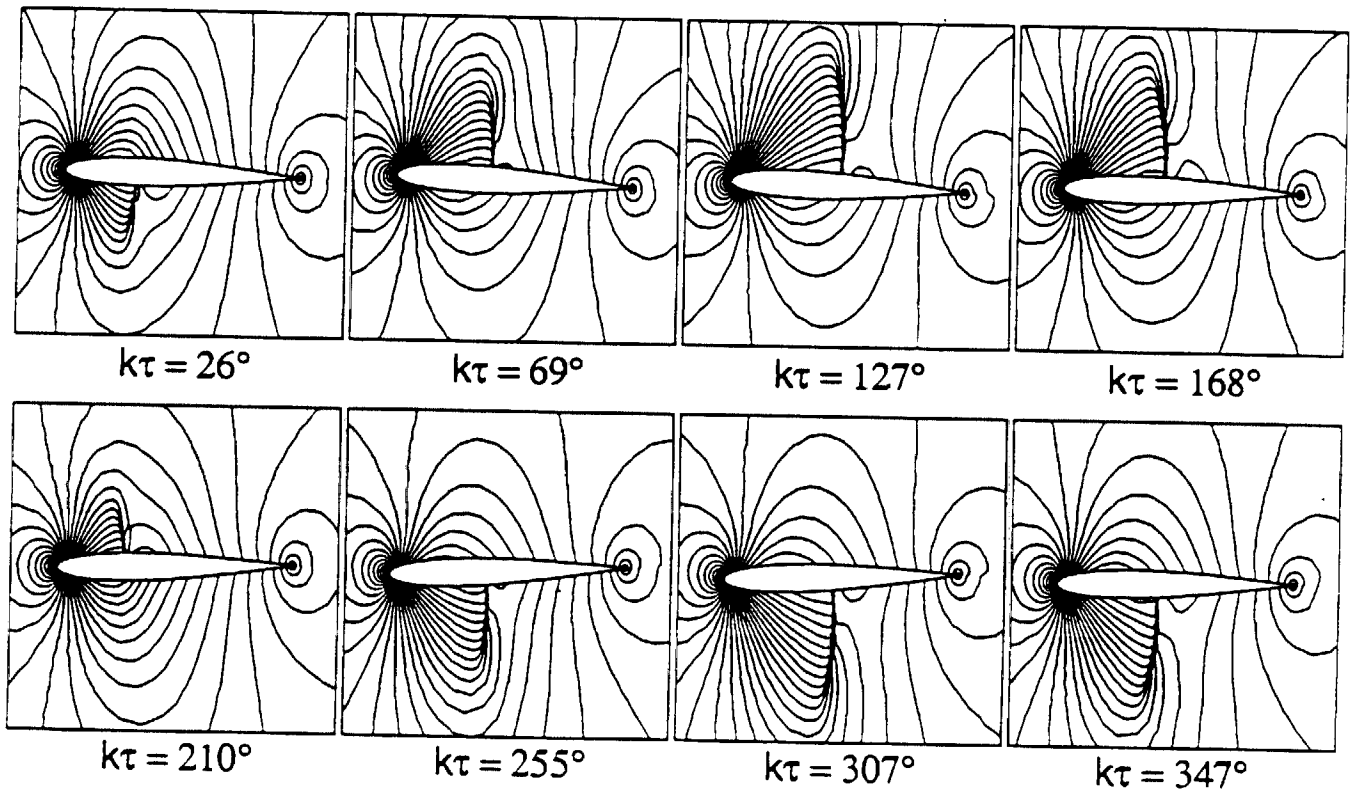
(b) final solution.

Fig. 13 Steady example of mesh enrichment for a 75° swept flat plate delta wing at $M_\infty = 1.4$, $\alpha = 20°$, and $\beta = 10°$ computed using the conical Euler equations.



(a) instantaneous meshes.

Fig. 14 Spatial adaption calculation for the NACA 0012 airfoil pitching harmonically at $M_\infty = 0.755$, $\alpha_0 = 0.016°$, $\alpha_1 = 2.51°$, and $k = 0.0814$ computed using an upwind-type Euler solution algorithm with flux-vector splitting.



(b) density contour lines.

Fig. 14 Concluded.

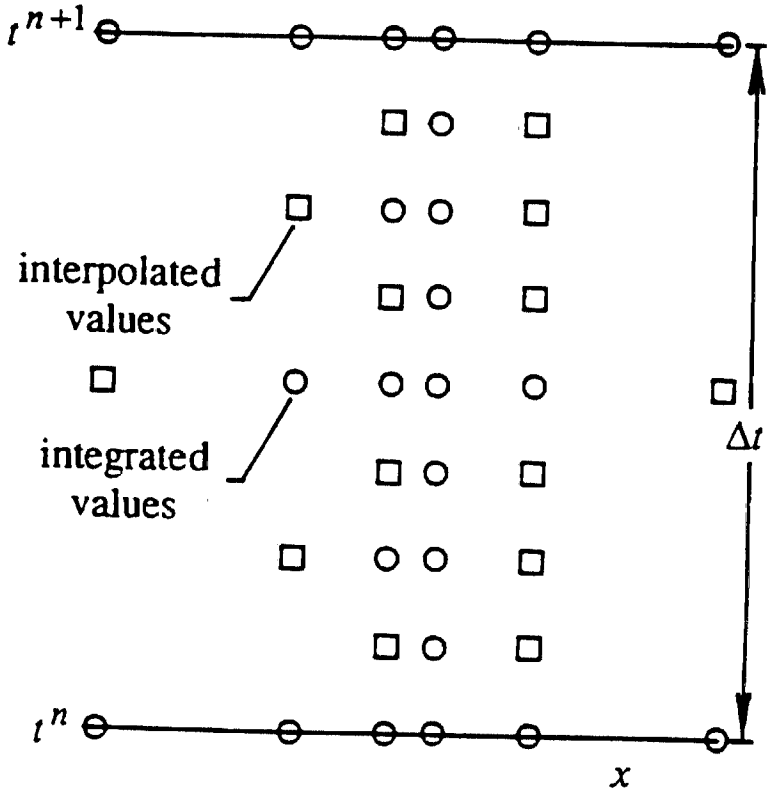


Fig. 15 Temporal stencil illustrating the details of the temporal adaption technique.

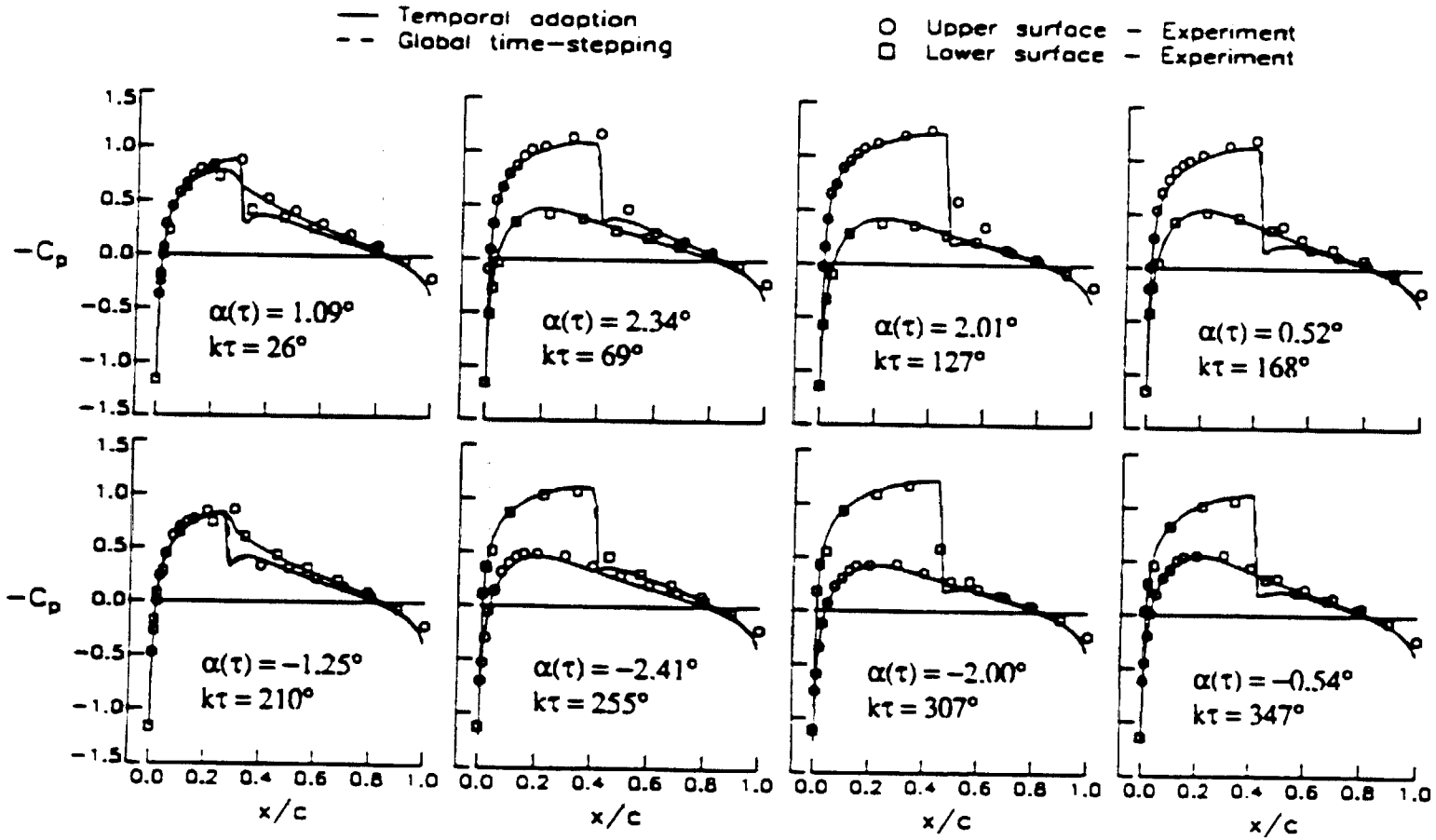


Fig. 16 Comparison of instantaneous pressure distributions for the NACA 0012 airfoil pitching harmonically at $M_\infty = 0.755$, $\alpha_0 = 0.016°$, $\alpha_1 = 2.51°$, and $k = 0.0814$ computed using an upwind-type Euler solution algorithm with flux-vector splitting.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	July 1992	Technical Memorandum

4. TITLE AND SUBTITLE

CFD Methods Development Considerations for Unsteady Aerodynamic Analysis

5. FUNDING NUMBERS

505-63-50-12

6. AUTHOR(S) John T. Batina

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
NASA Langley Research Center
Hampton, VA 23665-5225

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, DC 20546-0001

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

NASA TM 107644

11. SUPPLEMENTARY NOTES

Invited Paper Presented at the Workshop on Computational Aeroacoustics
Hampton, Virginia, April 6-9, 1992

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Unclassified - Unlimited
Subject Category 02

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The development of computational fluid dynamics (CFD) methods for unsteady aerodynamic analysis is described. Special emphasis is placed on considerations that are required for application of the methods to unsteady aerodynamic flow problems. Two broad categories of topics are discussed including grid considerations and algorithm development considerations, and example calculations are presented to illustrate the major points. Although the primary application of these CFD methods is to relatively low-frequency oscillatory phenomena such as flutter, the ideas that are presented may be of value to developers of computational aeroacoustics methods for predicting high-frequency acoustics.

14. SUBJECT TERMS

Computational Fluid Dynamics
Unsteady Aerodynamics
Aeroacoustics

15. NUMBER OF PAGES

30

16. PRICE CODE

A03

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	